# Multi-Depot Rural Postman Problems

Elena Fernández [*1] and Jessica Rodríguez-Pereira [†1]

[1]Department of Statistics and Operation Research, Universitat Politècnica de Catalunya-BcnTech, Spain

October 3, 2016

ABSTRACT

This paper studies Multi-Depot Rural Postman Problems on an undirected graph. These problems extend the well-known Undirected Rural Postman Problem to the case where there are several depots instead of just one. Linear integer programming formulations that only use binary variables are proposed for the problem that minimizes the overall routing costs and for the model that minimizes the length of the longest route. An exact branch-and-cut algorithm is presented for each considered model, where violated constraints of both types are separated in polynomial time. Despite the difficulty of the problems, the numerical results from a series of computational experiments with various types of instances illustrate a quite good behavior of the algorithms. When the overall routing costs are minimized, over 43% of the instances were optimally solved at the root node, and 95% were solved at termination, most of them with a small additional computational effort. When the length of the longest route is minimized, over 25% of the instances were optimally solved at the root node, and 99% were solved at termination.

## 1 Introduction

In this paper we present Multi-Depot Rural Postman Problems (MDRPPs) on undirected graphs. Similarly to other arc routing problems, in MDRPPs service demand is placed at a subset of edges, denoted demand or required edges. The distinguished feature of MDRPPs is that there are several depots instead of just one. In MDRPPs feasible solutions are given by sets of routes, each of them starting and ending at one of the depots, where each demand edge is traversed at least once by some route. MDRPPs involve two types of decisions: the allocation of the demand edges to the depots and the construction of the set of routes. We consider two different MDRPP models, which differ from each other in the objective function. The first model uses a min-cost objective where the goal is to determine a set of routes of minimum total cost and will be referred to as MC-MDRPP. The MC-MDRPP extends to several

[*]e.fernandez@upc.edu

[†]jessica.rodriguez@upc.edu

depots the well-known undirected Rural Postman Problem (RPP) introduced in [34], which considers one single depot.

The second model that we study uses a min-max objective where the goal is to minimize the length of the longest route, and will be referred to as MM-MDRPP. In contrast to the MC-MDRPP, which minimizes the overall routing costs, but may produce routes which are unbalanced in terms of their length, the MM-MDRPP can be suitable when balanced routes are sought. The MM-MDRPP is related to the min-max $K$-Rural Postman Problem (MM-$K$-RPP) that has been studied by several authors for different types of graphs [8, 9, 10, 11, 12, 36]. The MM-$K$-RPP is an uncapacitated arc routing problem, with one single depot and a fixed number of vehicles, $K$. Each vehicle must perform a tour starting and ending at the depot. The objective is to minimize the length of the longest among the $K$ routes. It clear that the undirected MM-$K$-RPP is a particular case of the MM-MDRPP, by considering $K$ depots co-located at the same vertex.

The motivation for studying MDRPPs comes not only from their theoretical interest, but also from their potential applications. Similarly to other arc routing problems, such applications appear in a wide variety of practical cases. Mail delivery, garbage collection, road maintenance or pipelines inspection are typical examples of real-life applications. When considering large application areas for the above examples, there is usually more than one depot from which service demand can be satisfied. Such depots may be vehicle stations, dump sites, replenishment points or relay boxes. For instance, in urban waste collection companies usually operate from multiple depots. A possibility for handling such problems is to decompose them in as many independent problems as depots, first allocating to the different depots demand sectors within smaller operating areas, and then finding optimal routes within each sector. Such solution strategy is indeed suboptimal, as it can be possible to obtain better solutions if a global approach is applied in which the allocation and routing decisions are jointly addressed.

The literature on Multi-Depot Arc Routing Problems (MDARP) is scarce. To the best of our knowledge, this is the first work where an exact algorithm based on a mixed integer linear programming (MILP) formulation is proposed to solve the MDRPP on an undirected graph. A directed MDARP has been considered in the recent work [20], where an exact branch-and-cut algorithm is proposed for a collaboration arc routing problem. Other than this, all previous work on MDARPs we are aware of focuses on capacitated arc routing problems with multiple depots (MDCARPs). Some theoretical aspects of MDCARPs are considered in [37]. The asymmetric multi-depot capacitated arc routing problem is studied in [28] where a new formulation and exact solution algorithm are presented. Heuristic methods have been proposed for both undirected and directed MDCARPs. Sequential heuristics for the undirected MDCARP are proposed in [2, 32, 33]: A cluster-first-route-second strategy, where the assignment of arcs to depots is established before designing the routes, is applied in [2], and a route-first-cluster-second strategy is used in [32, 33], where a single route is created first, which is later divided into smaller routes. Population based heuristics have also been used for solving MD-

CARPs. For the undirected case, two different ant colony strategies are presented in [27], and a hybrid genetic algorithm with perturbation that incorporates a local search, a replacement method, and a perturbation mechanism is proposed in [26]. The directed case is addressed in [38], where an evolutionary approach is presented, which takes advantage of the extensions of the classical heuristics for the Capacitated Arc Routing Problem [23].

Multi depot routing problems are indeed related to districting, where a set of clusters or districts that suitably partition the demand set is sought. The design of good districts at an strategic level, where demand points or edges are allocated to depots, allows finding efficient routes at each district at an operational in a later phase. There exists a rich districting literature, in relation to arc routing. In fact, some of the above referenced works stem from this relation. As an example, the heuristics of [32, 33] are devised as a second phases in districting design problems. Two recent works on districting for arc routing are [13, 21]. The interested reader is addressed to [30, 31] for further reading on this topic.

In this work we exploit properties and optimality conditions of MDRPPs, when stated on undirected graphs. These properties allow us to obtain MILP formulations where all variables are binary. In its turn, using binary variables only makes it possible to model parity constraints with an adaptation of the co-circuit inequalities [6]. The resulting formulations have two families of constraints of exponential size on the number of vertices of the input graph: one set for the parity of vertices and the usual set of connectivity constraints. As we will see, the constraints in each family can be separated exactly in polynomial time with adaptations of well-known algorithms. We propose an exact branch-and-cut algorithm where the linear programming (LP) relaxation of the formulation is reinforced with several families of valid inequalities. To analyze the behavior of the proposed algorithm, we have run a series of computational experiments for both the MC-MDRPP and the MM-MDRPP, with a set of benchmark instances adapted from well-known data sets used for other arc routing problems in the literature. The results of these experiments are presented and analyzed.

The reminder of this work is structured as follows. In Section 2 we define MDRPPs whereas in Section 3 we state some properties that will be exploited in the proposed formulations. The formulations for the MC-MDRPP and the MM-MDRPP are presented in Section 4, as well as some families of valid inequalities that we use to reinforce their LP relaxations. Section 5 describes the solution algorithm and discusses the procedures used to separate violated connectivity and parity inequalities. The computational experiments and the obtained results are described in Section 6. Section 7 concludes the work with some comments and directions for future research.

## 2 Multi-Depot Rural Postman Problems

MDRPPs are defined on an undirected connected graph $G = (V, E)$ with vertex set $V$, $|V| = n$, edge set $E$, $|E| = m$, set of depots $D \subset V$, and set of demand (required) edges $R \subset E$. Non-demand edges are denoted by $F = E \setminus R$. The connected components induced by demand edges are denoted by $C_k = (V_k, R_k)$, $k \in K$ so $R = \bigcup_{k \in K} R_k$. Let $V_R = \bigcup_{k \in K} V_k$. We assume that $E$ contains no edge connecting two depots, and no component has more than one depot, although it is possible that a component contains no depot, i.e. $|V_k \cap D| \leq 1$ for all $k \in K$. Let $c$ denote a non-negative real cost function defined on the edges of $G$.

Throughout we use the term *route* to denote a closed path, not necessarily simple, that starts and ends at the same depot $d \in D$. When the depot of the route needs to be explicit we say that the route is *rooted* at depot $d$. We say that a demand edge $e \in R$ is *served* by a route, if the route traverses $e$ at least once. As usual, the cost of a route is the sum of the costs of the edges in the route, where the cost of each edge is counted as many times as it is traversed in the route.

We will use the following notation. For any non-empty vertex subset $S \subset V$, $\delta(S) = \{e \in E | e = (u, v), u \in S, v \in V \setminus S\} = \delta(V \setminus S)$ is the set edges in the cut between $S$ and $V \setminus S$ and $\gamma(S) = \{e \in E | e = (u, v), u, v \in S \text{ or vice versa}\}$ the set of edges with both vertices in $S$. For a singleton $S = \{v\}$, with $v \in V$, we do not use the brackets and just write $\delta(v) \equiv \delta(\{v\})$. For $H \subset E$ we use $\delta_H(S) = \delta(S) \bigcap H$ and $\gamma_H(S) = \gamma(S) \bigcap H$. Furthermore, we will say that a vertex $v \in V$ is $H$-odd if $|\delta_H(v)|$ is odd; otherwise we will say that $v$ is $H$-even. Finally, we use the standard compact notation $f(A) \equiv \sum_{e \in A} f_e$ where $A \subseteq E$, and $f$ is a vector or a function defined on $E$. If $f$ is only defined on subset $B \subset E$, we use $f(A) \equiv f(A \cap B) \equiv \sum_{e \in A \cap B} f_e$.

In the reminder of this paper we make the following modeling assumption:

(H1) Demand edges in the same component can be served from different depots. The effect of this assumption is illustrated in Figure 1. Figure 1.a shows the input graph, which has two demand components and one depot in each of them ($v_1$ and $v_2$, respectively). Demand edges are represented by black lines while non demand edges are drawn in light grey. The numbers next to the edges indicate their costs. Figure 1.b shows the optimal solution when we impose that all demand edges in the same component are served from the same depot, whose total cost is 23 units. The route of depot $v_1$ (represented with straight lines), which serves the demand edges of $C_1$, consists of edges ($v_1$, A), (A, B), and (B, $v_1$). The route of depot $v_2$ (represented with doted lines), which serves the demand edges of $C_2$, consists of edges ($v_2$, E), (E, C), (C, D), (D, E), (E, F), and (F, $v_1$). Figure 1.c shows that a better solution, of value 19 units, can be obtained if we allow to serve demand edges in the same component from different depots. Now all demand edges of $C_1$ and some demand edges of $C_2$ are served in the route from depot $v_1$ defined by edges ($v_1$, A), (A, C), (C, E), (E, D), (D, B), and (B, $v_1$). The remaining demand edges of this component are served in the route from depot $v_2$, which consists of edges ($v_2$, E), (E, F), and (F, $v_2$).

The routes not necessarily have to be vertex-disjoint.

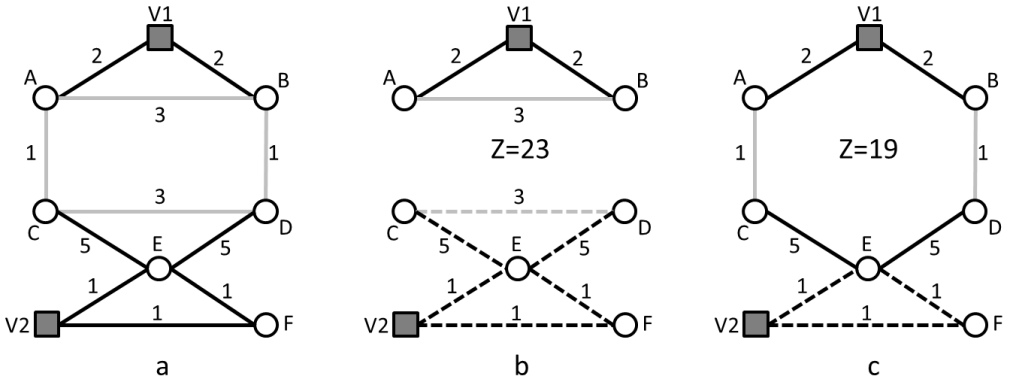



Figure 1: Example that allowing to split the demand components among routes may produce better solutions

In the following we assume that $G$ has been simplified so that $V$ is the set of vertices incident with edges in $R$, and $E$ contains the edges in $R$ plus additional non-demand edges, connecting every pair of vertices not connected with an edge of $R$, representing shortest paths in the original graph. For this, according to [14], first we add to $G_R = (V_R, R)$ an edge between every pair of vertices of $V_R$ having a cost equal to the shortest path length on $G$. Then we remove all non-demand edges $(i, j) \in F$ for which $c_{ij} = c_{ik} + c_{kj}$ for some $k \in V$ and one of two edges in parallel whenever they both have the same cost. Hence the costs satisfy the triangle inequality.

**Definition 2.1**

- *The MC-MDRPP is to find a set of routes that serve all the demand edges of minimum total travel cost.*

- *The MM-MDRPP is to find a set of routes that serve all the demand edges that minimizes the length of the longest route.*

# 3 Complexity and optimality conditions

The RPP is a particular case of the MC-MDRPP with $|D| = 1$. Since the RPP is NP-hard [35], we also have:

**Proposition 3.1** *The MC-MDRPP is NP-hard.*

The RPP is also a particular case of the MM-MDRPP with $|D| = 1$. Thus, we also have:

**Proposition 3.2** *The MM-MDRPP is NP-hard.*

As it is usual in other uncapacitated arc routing problems on undirected graphs the feasibility of MDRPP solutions is basically established via connectivity and parity conditions, in addition to the requirement that all demand edges are served. Thus it is not surprising that, similarly to other such problems with min-cost objectives, when non-negative costs satisfy the triangle inequality, optimality conditions hold for the MC-MDRPP that allow to derive formulations using binary variables (see, for instance, [19, 22]). These properties extend or can be adapted to the MM-MDRPP as well as explained below:

(O1) **MC-MDRPP and MM-MDRPP.** There is an optimal MDRPP solution in wich each demand edge is served by exactly one route.

(O2) **MC-MDRPP and MM-MDRPP.** There is an optimal solution in which no edge is traversed more than twice. Otherwise, two copies of the same edge can be removed without affecting neither the requirement that all demand edges are served, nor the parity of the vertices or the connectivity with the depot.

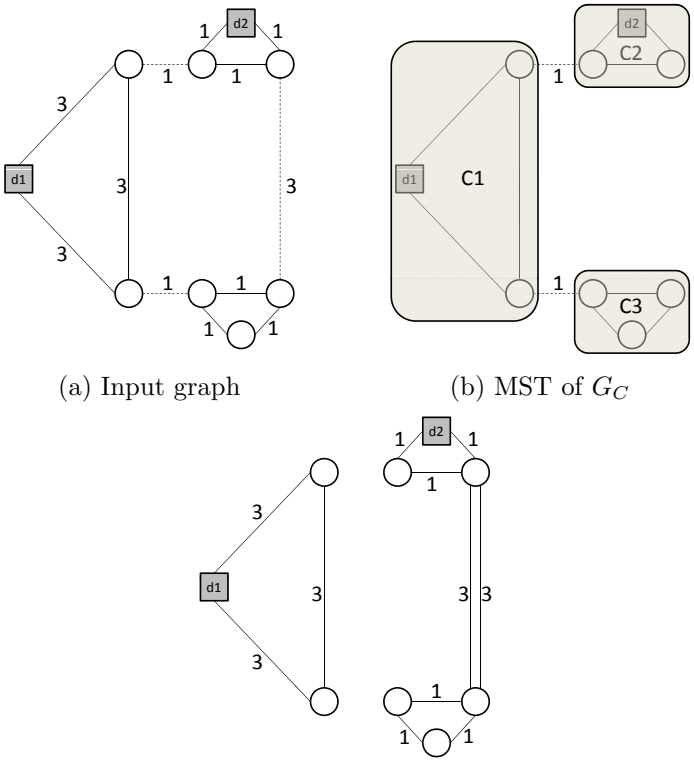
(O3) **MC-MDRPP and MM-MDRPP.** There is an optimal solution where no non-demand edge with the two end-nodes in the same component ($e \in \gamma_F(V_k)$) is traversed more than once. Otherwise, two copies of such an edge can be removed without affecting the feasibility of the solution. Furthermore, because of the triangle inequality, the only edges of $\gamma_F(V_k)$, that are used, are those connecting two $R$-odd vertices.

(O4) There is an optimal solution in which the only non-demand edges that are traversed twice are of one of the following types:

(a) **MC-MDRPP (see [22] for a similar result for the RPP).** Edges of the Minimum Spanning Tree (MST) of the multigraph graph $G_C = (V_C, E_C)$ induced by the connected components. $V_C$ contains a node representing each connected component $C_k$, $k \in K$. For each pair of distinct components $C_k$ and $C_{k'}$, $E_C$ contains an edge $(k_e, k'_e)$ associated with each original edge $e$ linking $C_k$ and $C_{k'}$, i.e. each edge $e \in \delta_F(V_k) \cap \delta_F(V_{k'})$, which inherits its cost from $G$.
It is clear that any MST of $G_C$ will use only least cost edges between pairs of components. Let $T^*$ be an MST of $G_C$, and suppose an edge $e^* \in E$ connecting components $C_k$ and $C_{k'}$ is traversed twice in an optimal MC-MDRPP solution $s^*$, but $(k_{e^*}, k'_{e^*})$ is not a least cost edge connecting components $C_k$ and $C_{k'}$. Then, adding edge $e^*$ to $T^*$ produces a cycle in $G_C$, in which $c_{\widehat{e}} < c_{e^*}$, where $\widehat{e}$ denotes a least cost edge in such cycle. Then, replacing in $s^*$ the two copies of edge $e^*$ by two copies of $\widehat{e}$ produces feasible solution: the parity of the vertices of the original graph $G$ does not change and the connectivity of the new solution is guaranteed by the two copies of $\widehat{e}$. It is possible that in the new solution some edges are served from a different depot than in the original solution $s^*$, but this does not affect to its feasibility either. The fact that the cost of the new solution is smaller than that of the original one, contradicts the optimality

of the original solution.
As shown in the example of Figure 2 this optimality condition does not apply to the MM-MDRPP, where an optimal solution may have two copies of a non-demand edge connecting two different components, which does not belong to any MST of $G_C$. Thus the adaptation of this condition to the MM-MDRPP must take into account all least cost edges connecting any pair of components.

(b) **MM-MDRPP.** Least cost edges connecting pairs of vertices of the multi-graph graph $G_C = (V_C, E_C)$.




(a) Input graph (b) MST of $G_C$

(c) Optimal solution

Figure 2: Optimal MM-MDRP solution using twice an edge not in the MST of $G_C$.

# 4 Formulations

Below we present a MILP formulation for the MC-MDRPP that exploits the optimality conditions of the previous section. Condition (O2) implies that we only need two sets of binary variables, for the first and second traversals of edges, respectively. We denote by $E^y \subset E$ the set of edges that can be traversed twice in an optimal MC-MDRPP solution, which consists of all demand edges plus the edges of the MST of $G_C$ (see conditions O4(a)). In each set, variables are also associated with the depot of the route that traverses the edges. For each $e \in E$, $d \in D$, let $x_e^d$ be a binary

variable indicating whether or not edge $e$ is traversed by the route of depot $d$. For each $e \in E^y$, $d \in D$, let $y_e^d$ be a binary variable that takes the value one if and only if edge $e$ is traversed twice in the route of depot $d$. Then, the a MILP for the MC-MDRPP is as follows:

$$\min \sum_{d \in D} \left( \sum_{e \in E} c_e x_e^d + \sum_{e \in E^y} y_e^d \right) \tag{1}$$

$$(x^d + y^d)(\delta(d)) \geq 2, \qquad d \in D \tag{2}$$

$$(x^d + y^d)(\delta(S)) \geq 2x_e^d, \qquad d \in D, S \subseteq V \setminus \{d\}, \; e \in \gamma(S) \tag{3}$$

$$(x^d - y^d)(\delta(S) \setminus H) + y^d(H) \geq x^d(H) - |H| + 1, \qquad S \subset V, \; H \subseteq \delta(S), \; |H| \text{ odd}, d \in D \tag{4}$$

$$\sum_{d \in D} x_e^d = 1, \qquad e \in R \tag{5}$$

$$y_e^d \leq x_e^d, \qquad e \in E^y, d \in D \tag{6}$$

$$x_e^d \in \{0,1\}, e \in E \qquad y_e^d \in \{0,1\}, e \in E^y \qquad d \in D \tag{7}$$

Inequalities (2) and (3) ensure that all depots are used and the connectivity of each route with its depot, respectively. This later condition is imposed by stating that if an edge is traversed by the route associated with depot $d \in D$, then at least two edges must cross the cut-set of any vertex set containing its two end-nodes, if it does not contain the depot $d$. Inequalities (4) ensure the parity (even degree) of every subset of vertices and, in particular, at every vertex. Broadly speaking, they impose that if a solution uses an odd number of edges, $H$, incident to a set of vertices $S$, then the solution uses at least one additional traversal of some edge in the cut-set $\delta(S)$. In our case, we further exploit the precedence relationship of the $x$ variables with respect to $y$ variables imposed by constraints (6). Thus, the additional edge will be either a second traversal of some edge of $H$ or a first traversal of some edge of $\delta(S) \backslash H$. Inequalities (4) are an adaptation to the MDRPP of those proposed in [3, 4, 5], which were later reinforced in [18] for the Maximum Benefit Chinese Postman Problem. Inequalities (2)-(4) jointly guarantee that any solution defines $|D|$ Eulerian circuits. Taking into account optimality condition (O1), equalities (5) ensure that each demand edge is served by one route. As mentioned, inequalities (6) impose that a route cannot traverse an edge for the second time unless it also traverses the edge for the first time. Binary conditions of the variables $x$ and $y$ derived from their definition are reflected in constraints (7).

The above formulation has $|E| \times |D|$ $x$ variables and $|E^y| \times |D|$ $y$ variables. There are $|D|$ inequalities of type (2), $|R|$ inequalities (5) and $|E^y| \times |D|$ inequalities of type (6). The size of the families inequalities (3) and (4) is exponential on $|V|$.

For the formulation for the MM-MDRPP we use the same sets of decision variables $x$ and $y$, taking into account that the index set $E^y$ for the variables associated

with edges that can be traversed twice in an optimal MM-MDRPP solution must be defined according to conditions O4(b). The formulation inherits all constraints (2)-(7), In addition we define a new integer variable $z$ representing the length of the longest route, so the objective becomes the minimization of $z$. A new family of constraints is needed to relate the new variable $z$ to the lengths of the routes. These inequalities, also ensure that $z$ represents the longest route:

$$z \geq \sum_{e \in E} c_e x_e^d + \sum_{e \in E^y} c_e y_e^d \qquad v \in D. \tag{8}$$

### 4.1 Valid Inequalities

Below we present some families of simple valid inequalities that we will use to reinforce the LP relaxations of the above formulations for the MC-MDRPP and the MM-MDRPP:

1. **Aggregated connectivity constraints.**
   By adding up over all depots the connectivity constraints (3) associated with subsets of nodes containing no depots, and taking into account that all vertices, except possibly the depots, are incident with some demand edge, and thus must be visited, we obtain:

   $$\sum_{d \in D} (x^d + y^d)(\delta(S)) \geq 2, \qquad S \subset V, S \cap D = \emptyset. \tag{9}$$

   Even if the family (9) is implied by the general family (3) and is also of exponential size, some small sub-families associated with particular subsets $S$ can be very useful to reinforce the initial LP relaxation when the general family (3) is relaxed:

   - Singletons $S = \{v\}$, with $v \in V \setminus D$:

     $$\sum_{d \in D} (x^d + y^d)(\delta(v)) \geq 2, \qquad v \in V \setminus D. \tag{10}$$

     For the depots $d \in D$ the inequalities (10) are also valid, although they are dominated by the stronger constraints (2).

   - End-nodes of demand edges. $S^e = \{u, v\}$, with $e = (u, v) \in R$, $S^e \cap D = \emptyset$:

     $$\sum_{d \in D} (x^d + y^d)(\delta(S^e)) \geq 2, \qquad e \in R, S^e \cap D = \emptyset. \tag{11}$$

   - Vertex sets of components without depot. $S = V_k$, $k \in K$, $V_k \cap D = \emptyset$:

     $$\sum_{d \in D} (x^d + y^d)(\delta(V_k)) \geq 2, \qquad k \in K, V_k \cap D = \emptyset. \tag{12}$$

2. **Aggregated parity constraints.** Aggregate versions of the parity constraints (4) are indeed valid for the MC-MDRPP and the MM-MDRPP. Similar inequalities (but combining binary and general integer variables) have been used for other ARPs with multiple vehicles, namely the MM-$K$-RPP [8, 12]. For the MC-MDRPP and the MM-MDRPP, for $S \subset V$, $H \subseteq \delta(S)$, $|H|$ odd. the inequality that we obtain is the following:

$$\sum_{d \in D}(x^d - y^d)(\delta(S) \setminus H) + \sum_{d \in D} y^d(H) \geq \sum_{d \in D} x^d(H) - |H| + 1. \qquad (13)$$

In particular, when $S$ is $R$-odd, i.e. $|\delta_R(S)|$ odd, and $H = \delta_R(S)$, the inequality (13) becomes

$$\sum_{d \in D}(x^d - y^d)(\delta_F(S)) + \sum_{d \in D} y^d(\delta_R(S)) \geq 1. \qquad (14)$$

# 5 Solution Algorithm

In this section, we present the solution algorithm that we use to solve the MDRPPs, based on the formulations proposed in Section 4. It is a branch-and-cut algorithm where an iterative Liner Programming (LP) based cutting plane algorithm is applied to the nodes of the enumeration tree. For this, the families of constraints of exponential size are relaxed and, at each iteration, inequalities violated by the current LP solution are separated. Such inequalities are iteratively incorporated to the current formulation and the reinforced formulation resolved. In our case the two families of constrains of exponential size are the connectivity and the parity constraints, (3) and (4), respectively. In the following subsections, the different elements of our algorithm are described: the initial formulation that is considered and the separation procedures for inequalities (3) and (4).

## 5.1 Initial relaxation

The algorithm starts with the integrality conditions relaxed and only a subset of constraints. Initially we include all constraints (2), (5) and (6), plus a small subset of connectivity and parity constraints. In particular, for each $d \in D$, we consider the inequalities (3) associated with the vertex subsets defined by the end-nodes of the edges not incident with any depot, i.e., $S^e = \{u, v\}$, with $e = (u, v) \in E$, such that $u, v \notin D$. This relaxation is reinforced with the aggregated connectivity inequalities (10), (11) and (12), plus the aggregated parity constraints (14), associated with $R$-odd singletons, i.e., $S = \{v\}$ with $v \in V$ and $|\delta_R(v)|$ odd.

## 5.2 Separation of inequalities

At any iteration of the algorithm the step that solves exactly the separation problem for the connectivity and parity inequalities consists of two parts. First, we look for

violated connectivity inequalities (3), and then we look for violated parity inequalities (4). Throughout $(\overline{x}, \overline{y})$ denotes the current LP solution and, for each depot $d \in D$, $(\overline{x}^d, \overline{y}^d)$ the partial LP solution associated with the depot $d$, i.e. the components of $(\overline{x}, \overline{y})$ associated with $d$. Furthermore, $G^d_{\overline{x},\overline{y}} = (V^d, E_{\overline{x}^d,\overline{y}^d})$ denotes the support graph of the partial solution $(\overline{x}^d, \overline{y}^d)$ for depot $d \in D$, obtained from $G$ by eliminating all edges in $E$ with $\overline{x}^d_e = 0$ and all vertices that are not incident with any edge of $E_{\overline{x}^d,\overline{y}^d}$.

#### 5.2.1 Separation of connectivity inequalities (3)

To separate the connectivity inequalities (3), for each depot $d \in D$, we first check if $G^d_{\overline{x},\overline{y}}$ is connected. If it is not, each connected component $C$ with vertex set $V(C) \subseteq V^d \setminus D$ defines a violated connectivity constraint (3) for depot $d$. When $G^d_{\overline{x},\overline{y}}$ is connected we build the tree of min-cuts $T^d$ (see, for instance, [24]) of $G^d_{\overline{x},\overline{y}}$ with capacities given by $\overline{x}^d_e + \overline{y}^d_e$. Then, we use an adaptation of the algorithm of [7]. For each edge $e = (u, v)$ in $E_{\overline{x}^d,\overline{y}^d}$ with $u, v \in V \setminus D$, the minimum cut $\delta(S)$ such that $e \in \gamma(S)$ is easily obtained from the min-cut tree $T^d$. If the value of the min cut is smaller than $2\overline{x}^d_e$ then the inequality (3) associated with $S$ and $d$ is violated by $(\overline{x}^d, \overline{y}^d)$. The above separation is exact and similar to the procedure used by other authors to separate constraints (3) for other arc routing problems [1, 4, 17].

#### 5.2.2 Separation of parity inequalities (4)

For a given vector $(\overline{x}, \overline{y})$, the separation problem for inequalities (4) is to find $d \in D$, $S \subset V$, $H \subseteq \delta(S)$, $|H|$ odd such that the associated parity inequality (4) is violated by $(\overline{x}, \overline{y})$, or to prove that no such inequality exists. The algorithm that we use follows the spirit of the separation used by other authors with similar parity constraints for other arc routing problems with binary variables [4, 5, 17]. For each $d \in D$, the algorithm starts by building the tree of min-cuts of the support graph $G^d_{\overline{x},\overline{y}}$, $T^b$, with capacities vector $b$ defined as follows. Each edge $e = (u, v) \in E_{\overline{x}^d,\overline{y}^d}$ is assigned a capacity given by $b_e = min\{(\overline{x}^d_e - \overline{y}^d_e), 1 - (\overline{x}^d_e - \overline{y}^d_e)\}$. This criterion dictates whether edge $e$ should be assigned to $H$ or to $\delta(S) \setminus H$ if the selected cut-set contains edge $e$, in order to obtain the smallest possible value of the left hand side of (4).

When $T^b$ has a cut $\delta(S)$ of capacity smaller than one, i.e. $b(\delta(S)) < 1$, we consider its vertex set $S$, and the set of edges $H = \{e \in \delta(S) \mid (\overline{x}^d_e - \overline{y}^d_e) \geq 0.5\}$, which, as explained, produces the smallest possible value on the left hand side of (4). When $|H|$ is odd, $H$ defines, together with $S$, a violated inequality of type (4). Otherwise, if $|H|$ is even, since all capacities are non-negative, the smallest increment in the value of the left hand side of (4) that guarantees that $|H|$ is odd is obtained by either removing one edge from $H$ (and transferring it to $\delta(S) \setminus H$) or by adding to $H$ one edge currently in $\delta(S) \setminus H$. By doing so the variation in $|H|$ is of just one unit, so the new set $H$ will be odd. The above mentioned smallest increment can be easily computed as

$$\Delta = min\left\{\min\{\overline{x}^d_e - \overline{y}^d_e : e \in \delta(S) \setminus H\}, \min\{1 - (\overline{x}^d_e - \overline{y}^d_e) : e \in H\}\right\},$$

where the first term represents the minimum increment in the left-hand-side if an edge of $\delta(S) \setminus H$ is transferred to $H$, and the second term represents the minimum increment if an edge of $H$ is transferred to $\delta(S) \setminus H$. When $b(\delta(S)) + \Delta < 1$, the updated set $H$ defines a violated inequality (4) for $d$ and $S$ for the current solution $(\overline{x}^d, \overline{y}^d)$.

It is possible that the minimum cut-set of $T^b$ does not produce a violated inequality (4) even it it exists. This is only possible if the set $H$ associated with the minimum cut in $T^b$ is even. Fortunately, in [29] it is proven that exploring as explained above all the cut-sets of $T^b$ defines an exact algorithm for knowing whether or not a violated inequality (4) exists. The order of such an algorithm is dominated by that of the algorithm that obtains the min-cut tree $T^b$. In practice, however, this upper limit on the order of the algorithm is very seldom reached. On the one hand, each connected component of $G^d_{\overline{x},\overline{y}}$ for the capacities vector $b$ already defines some of the subsets $S$ of the tree $T^b$ and connected components can be obtained with a small computational burden. On the other hand, when $G^d_{\overline{x},\overline{y}}$ defines one single connected component but a violated inequality exists, most often the cut-set producing the violated inequality will be identified before completing the full cut-tree $T^b$. Thus, in most cases, only if no violated inequality (4) exists it will be necessary to compute all the min-cuts that define $T^b$.

Summarizing, for a fixed depot $d \in D$, the exact separation for inequalities (4) reduces to finding the set $S$ such that $\delta(S)$ contains the best possible set $H$, and indicates that, in the worst case, this problem can be solved by finding the the complete tree of min-cuts of the support graph $G^d_{\overline{x},\overline{y}}$, for the capacities vector $b$ defined above. It is important to recall that the smallest value of the left hand side of inequality (4) after making $H$ odd is not necessarily associated with the smallest min-cut of the tree.

# 6 Computational Experiments

In this section, we describe the computational experiments we have run in order to evaluate the performance of the proposed algorithm. Programs have been coded in C++ using CPLEX 12.5 Concert Technology for the solution of the LP relaxations. The maximum computing time has been set to four hours. Moreover, the cuts generated by CPLEX have been disabled.

All the instances were run on an Intel Core 2 CPU, 2.67 GHz and 8.00 GB RAM. Since there were no available MDRPP benchmark instances, we have generated test instances with two and four depots from 118 well-known RPP benchmark instances. The original RPP benchmark instances are divided in five groups. The first group ALB contains two data sets ALBAIDAA and ALBAIDAB, obtained from the Albaida, Spain Graph (see [15, 16]). The second group contains the 24 instances, labeled P, of [14]. The last 3 groups contain instances from [25]: 36 instances with vertices of degree 4 and disconnected required edges sets (labeled D), 36 grid instances (labeled G), and 20 randomly generated instances (labeled R). In all cases we inherited the set of required edges and the cost function $c$ from the original RPP instances.

| | ♯ inst | $\|V_0\|$ | $\|E_0\|$ | $\|R\|$ | $\|K\|$ | $\|V\|/\|V_0\|$ | $\|E\|/\|E_0\|$ |
|---|---|---|---|---|---|---|---|
| ALB | 2 | 90-102 | 144-166 | 88-99 | 10-11 | 1.00 | 0.99 |
| P | 17 | 7-50 | 13-184 | 7-78 | 2-8 | 1.00 | 0.99 |
| D16 | 6 | 16 | 31 | 3-16 | 2-5 | 0.83 | 0.80 |
| D36 | 9 | 36 | 72 | 10-38 | 4-11 | 0.78 | 0.79 |
| D64 | 9 | 64 | 128 | 27-75 | 5-15 | 0.82 | 0.83 |
| D100 | 9 | 100 | 200 | 50-121 | 9-22 | 0.85 | 0.87 |
| G16 | 7 | 16 | 24 | 3-13 | 3-5 | 0.74 | 0.67 |
| G36 | 9 | 36 | 60 | 11-35 | 5-9 | 0.79 | 0.75 |
| G64 | 9 | 64 | 112 | 24-68 | 4-14 | 0.80 | 0.78 |
| G100 | 9 | 100 | 180 | 41-113 | 4-20 | 0.83 | 0.83 |
| R20 | 2 | 20 | 47-75 | 3-7 | 3-4 | 0.48 | 0.36 |
| R30 | 5 | 30 | 70-112 | 7-11 | 4-6 | 0.47 | 0.41 |
| R40 | 5 | 40 | 82-203 | 8-18 | 5-9 | 0.50 | 0.50 |
| R50 | 5 | 50 | 130-203 | 13-20 | 6-12 | 0.50 | 0.54 |

Table 1: Instances summary

Table 1 depicts information on these instances, which have been grouped according to their characteristics and sizes. The meaning of the columns is as follows: column under ♯ *inst* gives the number of instances in the group; columns under $|V_0|$ and $|E_0|$ give, respectively, the number of vertices and edges of the original graph; the columns under $|R|$ and $|K|$ give, respectively, the number of required edges and the number of connected components in the graph induced by those required edges. In the above columns, when not all the instances of the group had the same value, the minimum and maximum values of the group are given. The remaining columns in the table give information on the effect of the graph transformation. In particular, columns under $|V|/|V_0|$ and $|E|/|E_0|$ respectively correspond to the average ratios of the number of vertices or edges in the transformed graph related to the original graph. As it is known, the transformed graph is considerably smaller than the original graph, in terms of the number of vertices and edges.

For the computational experiments and regarding the set of depots, we have considered two different cases: two and four depots. Depots have been chosen randomly from the set of vertices, fulfilling that no connected component has more than one depot. For this, for each selected number of depots $|D| \in \{2, 4\}$ we have proceeded as follows. First, we randomly generate $|D|$ different numbers, $k_i$, $i = 1, \ldots, |D|$, from an integer uniform distribution $U[1, |K|]$, which give the indices of the clusters were the depots are located. Then, for each selected cluster, $k_i$ the index of the node of $V_{k_i}$ that becomes the depot is obtained by randomly generating a number $v_i$ from an integer uniform distribution $U[1, |V_{k_i}|]$. In order to compare the results obtained with 2 and 4 depots, the instances that have fewer than four connected components have been removed from the experiment. Finally, the experiments have been run with two groups of 103 instances each.

| | $\sharp opt_0$ | $gap_0$ | $cutsC_0$ | $cutsP_0$ | $\sharp opt$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
|---|---|---|---|---|---|---|---|---|---|---|
| ALB | 0/2 | 2.40 | 3568 | 153 | 2/2 | 0 | 6889.50 | 311 | 10 | 200.18 |
| P | 5/17 | 2.97 | 472.47 | 33.71 | 17/17 | 0 | 318.06 | 50.24 | 1.81 | 1.87 |
| D16 | 6/6 | 0 | 56.83 | 8.33 | - | - | - | - | 0 | 0.03 |
| D36 | 1/9 | 0.92 | 366.67 | 39.89 | 9/9 | 0 | 149 | 32.33 | 5.67 | 0.60 |
| D64 | 0/9 | 1.59 | 1635.22 | 80.56 | 9/9 | 0 | 1516.33 | 178.67 | 20.22 | 16.24 |
| D100 | 0/9 | 4.11 | 4392.78 | 135.56 | 8/9 | 0.70 | 26876.56 | 1483.89 | 376.67 | 2452.42 |
| G16 | 5/7 | 1.52 | 23.14 | 12.29 | 7/7 | 0 | 12.71 | 7.14 | 1.43 | 0.03 |
| G36 | 3/9 | 1.72 | 313.67 | 43.22 | 9/9 | 0 | 189.44 | 32.67 | 2.33 | 0.53 |
| G64 | 2/9 | 1.75 | 1474.89 | 93.89 | 9/9 | 0 | 7733.11 | 662.11 | 164.11 | 156.77 |
| G100 | 0/9 | 4.59 | 14368.89 | 422.44 | 7/9 | 2.80 | 59850.33 | 25381.78 | 337.44 | 4631.05 |
| R20 | 2/2 | 0 | 7.50 | 5.50 | - | - | - | - | - | 0.02 |
| R30 | 4/5 | 0.23 | 59.80 | 11.60 | 5/5 | 0 | 5.60 | 6.8 | 3 | 0.10 |
| R40 | 4/5 | 0.09 | 330.60 | 23.60 | 5/5 | 0 | 100.40 | 18 | 3.4 | 0.28 |
| R50 | 4/5 | 0.35 | 351.40 | 32.20 | 5/5 | 0 | 67.6 | 2.40 | 0.4 | 0.17 |

Table 2: Summary of results for MC-MDRPP for instances with two depots

### 6.1 Minimizing the overall routing costs: Results for MC-MDRPP

The results for the MC-MDRPP for the instances with two and four depots are summarized in Tables 2 and 3, respectively. For each group of instances, columns 2-5 give information about the root node of the enumeration tree, while columns 6-11 give the results of the search tree. Column under $\sharp opt_0$ shows the number of instances in the group that have been optimally solved in the root node. Column under $gap_0$ gives the average percentage gap at the root node with respect to the optimal or best-known solution at termination. The following two columns, under $cutsC_0$ and $cutsP_0$ give the average number of connectivity (3), and parity (4) cuts generated at the root node, respectively. Similarly, the next four columns under $\sharp opt$, $gap$, $cutsC$ and $cutsP$ give the same information at termination: number of instances that have been optimally solved, the average percentage gap with respect to the optimal or best-known solution and the average number of connectivity and parity cuts generated after the root node, respectively. Column under $\sharp Nod$ shows the average number of nodes that were explored in the search tree. Finally, the column under $cpu$ gives the overall computing time in seconds. These times do not include the preprocessing time for the reduction of the graph neither the time for loading the formulation, which are negligible as compared to the solution times reported in the tables. Detailed results for each instance can be found in the Appendix.

For 36 2-depot instances, a provable optimal solution was obtained already at the root node. At termination, optimality of the current solution was proven for 100 of the 103 2-depot instances. The unsolved instances are D35, G33 and G34 with percentage optimality gaps at termination of 6.34%, 9.88% and 15.36%, respectively.

For the 4-depot instances, optimality was proven at the root node for 53 instances and at termination for 95 of the 103 benchmark instances. No feasible integer solution was found within the time limit for any of the eight unsolved instances: two instances in group D100 (D34 - D35) and six instances in group G100 (G30 - G35).

As for the number of added inequalities, there were considerably more connec-

| | $\sharp opt_0$ | $gap_0$ | $cutsC_0$ | $cutsP_0$ | $\sharp opt$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
|---|---|---|---|---|---|---|---|---|---|---|
| ALB | 0/2 | 1.8 | 18263 | 429 | 2/2 | 0 | 45754 | 1325 | 129 | 5476.70 |
| P | 11/17 | 0.73 | 455.94 | 49.88 | 17/17 | 0 | 2103.06 | 134.65 | 1.81 | 44.77 |
| D16 | 6/6 | 0 | 0.50 | 1.33 | - | - | - | - | - | 0.01 |
| D36 | 4/9 | 0.87 | 488.89 | 69.67 | 9/9 | 0 | 219.33 | 35.22 | 1.89 | 0.96 |
| D64 | 1/9 | 2.27 | 4027 | 196.33 | 9/9 | 0 | 3509.78 | 1543.33 | 40.11 | 108.64 |
| D100 | 0/9 | 24.60 | 17860.11 | 480.11 | 7/9 | 22.22 | 51504.33 | 1878.44 | 130 | 7085.23 |
| G16 | 7/7 | 0 | 1.86 | 7.8 | - | - | - | - | - | 0.01 |
| G36 | 5/9 | 1.56 | 340.78 | 69.78 | 9/9 | 0 | 795.22 | 98.22 | 45.44 | 10.50 |
| G64 | 5/9 | 0.67 | 2636.67 | 196.89 | 9/9 | 0 | 16248.78 | 931.22 | 128 | 1835.31 |
| G100 | 1/9 | 67.08 | 12813.33 | 548.22 | 3/9 | 66.67 | 56311.78 | 1341.67 | 38.33 | 9640.11 |
| R20 | 2/2 | 0 | 1.5 | 7.5 | - | - | - | - | - | 0.02 |
| R30 | 4/5 | 1.63 | 21.80 | 14.40 | 5/5 | 0 | 29.40 | 7.40 | 1.20 | 0.08 |
| R40 | 4/5 | 0.13 | 352.60 | 42.20 | 5/5 | 0 | 20 | 4 | 1.80 | 0.35 |
| R50 | 3/5 | 0.48 | 533.60 | 62.60 | 5/5 | 0 | 81.20 | 13.60 | 0.80 | 0.45 |

Table 3: Summary of results for MC-MDRPP for instances with four depots

tivity cuts than parity cuts, although in some cases the number of added cuts was not very large.

The computational effort required for solving the instances to optimality, can be evaluated by the required solving computing times. In this sense, only 5 instances with two depots and 14 of instances with 4 depots required more than one hour (including those instances for which no feasible solution was found within the time limit). Moreover, in 82 2-depot and 76 4-depot instances, respectively, the optimal solution was found in less than 1 minute. If we compare the difficulty in solving instances with two and four depots in terms of the required computing times, we can see that the algorithm is, in general, faster when the instances have fewer depots.

Observe that the proposed algorithm was able to solve at the root node more 4-depot than 2-depot instances, even if the former involve a larger number of variables. If we analyze those instances we can find a pattern. In general, the solved instances belong to groups of small size and, also, they have a small number of connected components.

The results of the computational experiments also allow us to observe whether in the optimal solutions, connected components are fully served by the same depot. Our results indicate that connected components are split in 22 of the 2-depot instances and in 30 of the 4-depot ones.

## 6.2 Balancing the length of the routes: Results for MM-MDRPP

The analysis of the solutions structures shows that in some instances the routes produced by MC-MDRPP are unbalanced in terms of their length. Below we present the results of a new series of computational experiments have been run with the formulation for the MM-MDRPP. For these experiments we have considered the 78 2-depot instances with up to 50 nodes, and the set of 60 4-depot instances that consists of all instances with up to 40 nodes plus the R50 set. For the excluded instances, in most cases, it was not possible to solve them to optimality and percentage optimality

| | $\sharp opt_0$ | $gap_0$ | $cutsC_0$ | $cutsP_0$ | $\sharp opt$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
|---|---|---|---|---|---|---|---|---|---|---|
| P | 1/12 | 8.31 | 319.92 | 27.75 | 12/12 | 0 | 678.58 | 156.42 | 35.58 | 4.43 |
| D16 | 3/6 | 3.97 | 69.50 | 17.17 | 6/6 | 0 | 35.17 | 21.67 | 5.17 | 0.14 |
| D36 | 0/9 | 12.59 | 448.78 | 41.78 | 9/9 | 0 | 1310.44 | 193.11 | 44.44 | 4.66 |
| D64 | 0/9 | 11.21 | 2084.78 | 88.33 | 8/9 | 0.11 | 17340.33 | 1484.22 | 1688.44 | 1866.19 |
| G16 | 4/7 | 9.71 | 38.43 | 13.86 | 7/7 | 0 | 18.43 | 13 | 3.71 | 0.04 |
| G36 | 2/9 | 6.12 | 379.67 | 43.78 | 9/9 | 0 | 1279.56 | 224.56 | 27.78 | 5.25 |
| G64 | 0/9 | 7.82 | 2720.33 | 122.44 | 9/9 | 0 | 25627.78 | 2532.00 | 624.56 | 1665.05 |
| R20 | 1/2 | 13.11 | 18 | 5 | 2/2 | 0 | 55.50 | 23.50 | 7 | 0.16 |
| R30 | 1/5 | 10.00 | 112.20 | 16.40 | 5/5 | 0 | 92.40 | 10.80 | 3.40 | 0.15 |
| R40 | 2/5 | 3.47 | 418.20 | 23.40 | 5/5 | 0 | 650 | 109.20 | 59 | 2.42 |
| R50 | 0/5 | 18.49 | 538.00 | 25.20 | 5/5 | 0 | 2749.40 | 197.60 | 55.80 | 1.41 |

Table 4: Summary of results for MM-MDRPP for instances with two depots

| | $\sharp opt_0$ | $gap_0$ | $cutsC_0$ | $cutsP_0$ | $\sharp opt$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
|---|---|---|---|---|---|---|---|---|---|---|
| P | 2/12 | 14.24 | 661.83 | 69.58 | 12/12 | 0 | 4578.58 | 661.67 | 1107 | 1227.46 |
| D16 | 4/6 | 5.60 | 38.83 | 21.50 | 6/6 | 0 | 0 | 1.33 | 1.17 | 0.08 |
| D36 | 1/9 | 20.06 | 1153.56 | 122.33 | 9/9 | 0 | 319.56 | 490.78 | 176.56 | 47.03 |
| G16 | 6/7 | 3.17 | 11.86 | 13.43 | 7/7 | 0 | 0 | 0 | 0.14 | 0.02 |
| G36 | 2/9 | 15.90 | 805.78 | 124.78 | 9/9 | 0 | 3662.22 | 679.78 | 295.22 | 1117.09 |
| R20 | 2/2 | 0 | 19.50 | 16 | - | - | - | - | - | 0.05 |
| R30 | 3/5 | 6.73 | 144.20 | 30.20 | 5/5 | 0 | 82 | 24.20 | 15.50 | 0.38 |
| R40 | 0/5 | 31.09 | 638.20 | 61.80 | 5/5 | 0 | 2370.40 | 306.40 | 120 | 30.25 |
| R50 | 0/5 | 36,78 | 1005.60 | 92.80 | 5/5 | 0 | 5808.60 | 1152,60 | 815.2 | 113.81 |

Table 5: Summary of results for MM-MDRPP for instances with four depots

gaps at termination were quite big.

The results of the new experiments are summarized in Tables 4 and 5, where columns have the same meaning as before. It can be seen that, in general, the gap at the root node, the number of cuts, the number of explored nodes and the computing times are worse than with the MC-MDRPP. Still, the proposed algorithm found the optimal solution for all the tested MM-MDRPP instances but one, the exception being instance D26.

Comparing the optimal solutions to both models, we observe that, as could be expected, the overall routing costs are, in general, higher in optimal MM-MDRPP solutions than in optimal MC-MDRPP solutions. Even if there are 19 2-depot instances and 20 4-depot instances where the overall length of all routes is the same in both models, the average overall length increase is 13.09% for the 2-depot instances and 21.14% for the 4-depot instances. The maximum increases are 52.80% in instance R17 with two depots, and 73.69% in instance R11 with four depots.

Nevertheless, we can also observe that when using model MM-MDRPP, the length of the maximum route usually decreases noticeably with respect to the maximum length in an optimal MC-MDRPP solution. Even if there are 13 2-depot instances and 21 4-depot instances where the length of the longest route does not change, on average the length of the maximum route decreases in 19% for the 2-depot instances and 27.20% for the 4-depot instances. The maximum decreases are

| | $\sharp opt_0$ | $gap_0$ | $cutsC_0$ | $cutsP_0$ | $\sharp opt$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
|---|---|---|---|---|---|---|---|---|---|---|
| P | 0/12 | 24.29 | 788.33 | 39.58 | 12/12 | 0 | 2641.25 | 369.33 | 286.00 | 25.43 |
| D16 | 0/6 | 30.80 | 125.00 | 17.67 | 6/6 | 0 | 317.50 | 117.83 | 52.17 | 0.55 |
| D36 | 0/9 | 21.49 | 639.78 | 40.44 | 9/9 | 0 | 5104.67 | 703.89 | 858.00 | 1685.29 |
| G16 | 1/7 | 30.43 | 67.86 | 18.29 | 7/7 | 0 | 135.43 | 37.86 | 27.14 | 0.23 |
| G36 | 0/9 | 24.89 | 692.11 | 49.56 | 9/9 | 0 | 4032.22 | 556.00 | 252.67 | 368.37 |
| R20 | 0/2 | 33.48 | 47.50 | 15.00 | 2/2 | 0 | 133.50 | 34.00 | 18.00 | 0.31 |
| R30 | 0/5 | 22.35 | 147.60 | 21.40 | 5/5 | 0 | 473.80 | 77.40 | 57.80 | 0.88 |
| R40 | 0/5 | 14.74 | 432.00 | 30.00 | 5/5 | 0 | 4079.40 | 441.60 | 414.80 | 28.81 |

Table 6: Summary of results for MM-$K$-RPP for instances with two routes

| | $\sharp opt_0$ | $gap_0$ | $cutsC_0$ | $cutsP_0$ | $\sharp opt$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
|---|---|---|---|---|---|---|---|---|---|---|
| P | 0/12 | 37.57 | 1812.50 | 93.58 | 10/12 | 12.22 | 14981.33 | 1822.83 | 3422.25 | 3682.92 |
| D16 | 0/6 | 48.76 | 236.17 | 42.50 | 6/6 | 0 | 1013.67 | 446.83 | 594.83 | 15.40 |
| D36 | 0/9 | 36.27 | 2132.33 | 167.78 | 7/9 | 7.10 | 20157.67 | 2088.78 | 11910.89 | 3871.17 |
| G16 | 1/7 | 32.54 | 153.86 | 57.57 | 7/7 | 0 | 301 | 112.86 | 72.71 | 2.10 |
| G36 | 0/9 | 47.21 | 2974.44 | 204 | 8/9 | 3.54 | 14838.67 | 1694 | 1127 | 4502.25 |
| R20 | 0/2 | 50.43 | 257.50 | 31.50 | 2/2 | 0 | 812.50 | 625 | 692.50 | 20.09 |
| R30 | 0/5 | 47.38 | 448.80 | 47.40 | 5/5 | 0 | 2249.80 | 478.80 | 442.20 | 44.59 |
| R40 | 0/5 | 40.99 | 1090 | 75.80 | 2/5 | 6.60 | 19397.40 | 2650 | 4209.40 | 3336.25 |

Table 7: Summary of results for MM-$K$-RPP for instances with four routes

46.15% in instance G25 with two depots, and 64.71% in instance G16 with four depots.

### 6.3 Balancing the length of the routes from one single depot

In this section we present the results of the last series of experiments we have run. They correspond to the undirected MM-$K$-RPP, which considers $K$ vehicles located at a one single depot and minimizes the length of the longest route. As mentioned in the introduction the MM-$K$-RPP is a particular case of the MM-MDRPP, by considering $K$ depots co-located at the same vertex and performing one single route from each co-located depot. For our experiments, we have considered all instances with up to 40 nodes. For each instance, we have randomly selected one of the depots, which has been replicated $K$ times, and all other previous depots have been ignored.

Tables 6 and 7 give the results obtained for $K \in \{2, 4\}$. Comparing optimal solutions to the MM-$K$-RPP and the MM-MDRPP, we observe that the cost of the longest route in optimal MM-$K$-RPP solutions increases considerably in comparison to the MM-MDRPP. Consequently, the total routing cost increases as well. The average maximum length increases in 26.84% for the 2-depot instances and in 102% for the 4-depot instances. This represents a total increase of overall length of 31.29 % and 116.50% respectively, on average.

The computational effort required for solving the MM-$K$-RPP instances to optimality is higher than the previous experiments, for instances of the same size and

characteristics. In comparison with the MM-MDRPP, the computing time of the MM-$K$-RPP increases around 1604% for 2-depot instances and 644% for 4-depot instances. Furthermore, eight 4-depots instances could not be optimality solved within the limit time. We attribute this increase in the difficulty for optimally solving the instances to the symmetry that now appears for the routes, as they can now be interchanged.

# 7 Conclusions

In this work we have introduced Multi-Depot Rural Postman Problems on undirected graphs and studied some of its properties and dominance relations for two variants of the problem. In one model the objective is to minimize the overall routing costs, whereas the second model uses a min-max objective function aiming at minimizing the length of the longest route. MILP formulations, which only use binary variables, have been presented for both models where, as usual, the families of constraints that enforce connectivity and parity of solutions are of exponential size. We have proposed an exact branch-and-cut algorithm where the separation problem of both families is solved exactly with adaptations of well-know algorithms. The performance of the algorithm has been tested for the two proposed models. For this, in each case we have solved two sets of instances, with two and four depots, respectively, with 103 instances each of them. For the min-cost objective, 35% and 51% of the instances of the first and second sets, respectively, were optimally solved at the root node. At termination these percentages raise to 97% and 92% of instances optimally solved, for the first and second set, respectively. Regardless the difficulty of the problem, the results produced by the algorithm are quite satisfactory for the model with the min-cost objective. The numerical experiments with the model where the objective is the minimization of the longest route, indicate that computationally the new formulation becomes notably more demanding. Nevertheless, the formulation for this min-max version is indeed successful in producing balanced routes.

The proposed algorithm produces, in general, quite good results for both models although that its performance decreases as the number of depots increase. The reason is that the number of variables of the proposed formulations becomes too high so as to efficiently solve medium size instances. For both models, the performance algorithm could be enhanced with the inclusion of some ad-hoc heuristic.

# Acknowledgements

This research has been partially supported by the Spanish Ministry of Economy and Competitiveness end EDRF funds through grants BES-2013-063633, and MTM2015-63779-R (MINECO/FEDER). This support is gratefully acknowledged.

## References


[1] D. Ahr. Contributions to multiple postmen problems. Technical report, PhD dissertation, Departament of Computer Science, Heidelberg University, Heidelberg, Germany, 2004.

[2] A. Amberg, W. Domschke, and S. Voß. Multiple center capacitated arc routing problems: A tabu search algorithm using capacitated trees. *European Journal of Operational Research*, 124(2):360–376, 2000.

[3] J. Aráoz, E. Fernández, and C. Zoltan. Privatized rural postman problems. *Computers & Operations Research*, 33(12):3432–3449, 2006.

[4] J. Aráoz, E. Fernández, and C. Franquesa. The clustered prize-collecting arc routing problem. *Transportation Science*, 43(3):287–300, 2009.

[5] J. Aráoz, E. Fernández, and O. Meza. Solving the prize-collecting rural postman problem. *European Journal of Operational Research*, 196(3):886–896, 2009.

[6] F. Barahona and M. Grötschel. On the cycle polytope of a binary matroid. *Journal of Combinatorial Theory, Series B*, 40(1):40–62, 1986.

[7] J.-M. Belenguer and E. Benavent. The capacitated arc routing problem: Valid inequalities and facets. *Computational Optimization and Applications*, 10(2): 165–187, 1998.

[8] E. Benavent, A. Corberán, I. Plana, and J. Sanchis. Min-max $k$-vehicles windy rural postman problem. *Networks*, 54:216–226, 2009.

[9] E. Benavent, A. Corberán, and J. Sanchis. A metaheuristic for the min-max windy rural postman problem with $k$-vehicles. *Computationqal Management Science*, 7:269–287, 2010.

[10] E. Benavent, A. Corberán, I. Plana, and J. Sanchis. New facets and enhanced branch-and-cut for the min-max $k$-vehicles windy rural postman problem. *Networks*, 58:255–272, 2011.

[11] E. Benavent, A. Corberán, G. Desaulniers, F. Lessard, I. Plana, and J. Sanchis. A branch-and-cut for the maximum benefit chinese postman problem. *Networks*, 63:34–45, 2013.

[12] E. Benavent, A. Corberán, I. Plana, and J. Sanchis. Arc routing problems with min-max objectives. In A. Corberán and G. Laporte, editors, *Arc Routing: Problems, Methods and Applications*, pages 255–280. MOS-SIAM Series on Optimization, Philadelphia, 2014.

[13] A. Butsch, J. Kalcsics, and G. Laporte. Districting for arc routing. *INFORMS Journal on Computing*, 26(4):809–824, 2014.

[14] N. Christofides, V. Campos, A. Corberán, and E. Mota. An algorithm for the rural postman problem. Technical report, Imperial College Report IC.O.R.81.5, 1981.

[15] A. Corberán and J. Sanchis. A polyhedral approach to the rural postman problem. *European Journal of Operational Research*, 79(1):95–114, 1994.

[16] A. Corberán and J. Sanchis. The general routing problem polyhedron: Facets from the rpp and gtsp polyhedra. *European Journal of Operational Research*, 108(3):538–550, 1998.

[17] A. Corberán, E. Fernández, C. Franquesa, and J. Sanchis. The windy clustered prize-collecting arc routing problem. *Transportation Science*, 45(3):317–344, 2011.

[18] A. Corberán, I. Plana, A. Rodríguez-Chía, and J. Sanchis. A branch-and-cut for the maximum benefit chinese postman problem. *Mathematical Programming*, 141(1):21–48, 2013.

[19] E. Fernández, O. Meza, R. Garfinkel, and M. Ortega. On the undirected rural postman problem: Tight bounds based on a new formulation. *Operations Research*, 51(2):281–291, 2003.

[20] E. Fernández, D. Fontana, and G. Speranza. On the collaboration uncapacitated arc routing problem. *Computers & Operations Research*, 67:120–131, 2016.

[21] G. García Ayala, J. González-Velarde, R. Ríos-Mercado, and E. Fernández. A novel model for arc territory design: Promoting eulerian districts. *International Transactions in Operations Research*, 23:433–458, 2015.

[22] G. Ghiani and G. Laporte. A branch-and-cut algorithm for the undirected rural postman problem. *Mathematical Programming*, 87(3):467–481, 2000.

[23] B. L. Golden and R. T. Wong. Capacitated arc routing problems. *Networks*, 11(3):305–315, 1981.

[24] D. Gusfield. Very simple methods for all pairs network flow analysis. *SIAM Journal on Applied Mathematics*, 19(1):143–555, 1993.

[25] A. Hertz, G. Laporte, and P. Nanchen. Improvement procedure for the undirected rural postman problem. *INFORMS Journal on Computing*, 11:53–62, 1999.

[26] H. Hu, T. Liu, N. Zhao, Y. Zhou, and D. Min. A hybrid genetic algorithm with perturbation for the multi-depot capacitated arc routing problem. *Journal of Applied Sciences*, 13(16):32–39, 2013.

[27] A. Kansou and A. Yassine. A two ant colony approaches for the multi-depot capacitated arc routing problem. In *Computers & Industrial Engineering, 2009. CIE 2009. International Conference on*, pages 1040–1045. IEEE, 2009.

[28] D. Krushinsky and T. van Woensel. An approach to the asymmetric multi-depot capacitated arc routing problem. *European Journal of Operational Research*, 244(1):100–109, 2015.

[29] A. N. Letchford and J.-J. Salazar-González. Stronger multi-commodity flow formulations of the capacitated vehicle routing problem. *European Journal of Operational Research*, 244(3):730–738, 2015.

[30] L. Muyldermans. Routing, districting and location for arc traversal problems. Technical report, PhD dissertation, Catholic University of Leuven, Leuven, Belgium, 2003.

[31] L. Muyldermans and G. Pang. Variants of the capacitated arc routing problem. In A. Corberán and G. Laporte, editors, *Arc Routing: Problems, Methods and Applications*, pages 223–254. MOS-SIAM Series on Optimization, Philadelphia, 2014.

[32] L. Muyldermans, D. Cattrysse, D. V. Oudheusden, and T. Lotan. Districting for salt spreading operations. *European Journal of Operational Research*, 139 (3):521–532, 2002.

[33] L. Muyldermans, D. Cattrysse, and D. Van Oudheusden. Districting for arc-routing applications. *Journal of the Operational Research Society*, 54:1209–1221, 2003.

[34] C. S. Orloff. A fundamental problem in vehicle routing. *Networks*, 4(1):35–64, 1974.

[35] C. S. Orloff. On general routing problems: Comments. *Networks*, 6(3):281–284, 1976.

[36] E. J. Willemse and J. Joubert. Applying min-max $k$ postmen problems to the routing of security guards. *The Journal of the Operational Research Society*, 63:245–260, 2012.

[37] S. Wøhlk. Contributions to arc routing. Technical report, PhD dissertation, University of Southern Denmark, Denmark, 2004.

[38] L. Xing, P. Rohlfshagen, Y. Chen, and X. Yao. An evolutionary approach to the multidepot capacitated arc routing problem. *Evolutionary Computation, IEEE Transactions on*, 14(3):356–374, 2010.

# Appendix

### MC-MDRPP results

### MM-MDRPP results

| | 2 Depots | | | | | | | | 4 Depots | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
| ALBA | 1.39 | 4622 | 211 | 0 | 10806 | 268 | 8 | 258.98 | 2.26 | 15239 | 347 | 0 | 83422 | 2375 | 239 | 10398.51 |
| ALBB | 3.40 | 2513 | 95 | 0 | 2973 | 354 | 12 | 141.38 | 1.34 | 21286 | 510 | 0 | 8086 | 275 | 19 | 554.88 |
| P1 | 0 | 14 | 0 | - | - | - | - | 0.00 | 0 | 0 | 0 | - | - | - | - | 0.00 |
| P2 | 0 | 43 | 6 | - | - | - | - | 0.01 | 0 | 0 | 0 | - | - | - | - | 0.01 |
| P3 | 7.58 | 48 | 27 | 0 | 451 | 62 | 13 | 0.59 | 0 | 303 | 69 | - | 0 | 8 | 2 | 0.50 |
| P5 | 26.81 | 43 | 2 | 0 | 278 | 103 | 50 | 1.04 | 0 | 66 | 27 | - | - | - | - | 0.08 |
| P6 | 2.40 | 238 | 44 | 0 | 311 | 69 | 15 | 0.45 | 1.76 | 596 | 93 | 0 | 135 | 18 | 3 | 0.94 |
| P10 | 0 | 25 | 22 | - | - | - | - | 0.02 | 0 | 0 | 8 | - | - | - | - | 0.03 |
| P14 | 4.02 | 279 | 29 | 0 | 272 | 59 | 7 | 0.70 | 0 | 102 | 25 | - | - | - | - | 0.08 |
| P15 | 0.49 | 181 | 17 | 0 | 8 | 8 | 3 | 0.11 | 0 | 94 | 25 | - | - | - | - | 0.17 |
| P16 | 2.50 | 191 | 26 | 0 | 94 | 40 | 3 | 0.30 | 0 | 215 | 43 | - | - | - | - | 0.16 |
| P17 | 0 | 196 | 20 | - | 0 | 0 | 1 | 0.11 | 0 | 13 | 10 | - | - | - | - | 0.08 |
| P18 | 0 | 135 | 28 | - | - | - | - | 0.14 | 2.85 | 255 | 52 | 0 | 23 | 8 | 2 | 0.27 |
| P19 | 0.06 | 229 | 40 | 0 | 7 | 6 | 1 | 0.39 | 3.72 | 75 | 6 | 0 | 332 | 52 | 6 | 0.44 |
| P20 | 1.95 | 1135 | 64 | 0 | 360 | 118 | 10 | 5.84 | 0.08 | 1887 | 127 | 0 | 4 | 20 | 1 | 2.26 |
| P21 | 1.38 | 1292 | 66 | 0 | 436 | 109 | 11 | 5.10 | 0 | 677 | 93 | - | - | - | - | 0.84 |
| P22 | 1.70 | 1331 | 45 | 0 | 3042 | 265 | 26 | 14.10 | 2.67 | 1218 | 76 | 0 | 33664 | 2067 | 461 | 748.46 |
| P23 | 0.43 | 1915 | 88 | 0 | 0 | 2 | 2 | 2.18 | 0 | 1504 | 126 | - | - | - | - | 1.59 |
| P24 | 1.20 | 737 | 40 | 0 | 148 | 13 | 3 | 0.70 | 1.29 | 746 | 68 | 0 | 1594 | 116 | 5 | 5.12 |

Table 8: MC-MDRPP results for instances of the group ALBAIDA and P

| | 2 Depots | | | | | | | | 4 Depots | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
| D02 | 0 | 55 | 9 | - | - | - | 0 | 0.03 | 0 | 0 | 4 | - | - | - | 0 | 0.02 |
| D04 | 0 | 29 | 8 | - | - | - | 0 | 0.03 | 0 | 0 | 0 | - | - | - | 0 | 0.00 |
| D05 | 0 | 53 | 7 | - | - | - | 0 | 0.03 | 0 | 0 | 0 | - | - | - | 0 | 0.00 |
| D06 | 0 | 33 | 6 | - | - | - | 0 | 0.02 | 0 | 3 | 4 | - | - | - | 0 | 0.03 |
| D07 | 0 | 10 | 0 | - | - | - | 0 | 0.02 | 0 | 0 | 0 | - | - | - | 0 | 0.01 |
| D08 | 0 | 161 | 20 | - | - | - | 0 | 0.05 | 0 | 0 | 0 | - | - | - | 0 | 0.00 |
| D09 | 0.21 | 119 | 15 | 0 | 0 | 0 | 1 | 0.09 | 0 | 111 | 23 | - | - | - | - | 0.09 |
| D10 | 0 | 86 | 27 | - | - | - | - | 0.08 | 0 | 64 | 28 | - | - | - | - | 0.06 |
| D11 | 0.55 | 166 | 26 | 0 | 11 | 10 | 1 | 0.14 | 0.54 | 899 | 88 | 0 | 6 | 18 | 1 | 0.69 |
| D12 | 0.36 | 245 | 43 | 0 | 70 | 71 | 12 | 0.39 | 2.28 | 106 | 33 | 0 | 32 | 21 | 2 | 0.14 |
| D13 | 0.58 | 302 | 42 | 0 | 29 | 20 | 1 | 0.27 | 2.61 | 413 | 81 | 0 | 109 | 43 | 6 | 0.80 |
| D14 | 0.05 | 779 | 66 | 0 | 0 | 2 | 1 | 1.06 | 0 | 597 | 66 | - | - | - | - | 0.30 |
| D15 | 1.09 | 568 | 46 | 0 | 12 | 24 | 2 | 0.41 | 1.20 | 615 | 94 | 0 | 50 | 15 | 1 | 0.83 |
| D16 | 4.17 | 503 | 52 | 0 | 949 | 130 | 30 | 2.20 | 0 | 711 | 114 | - | - | - | - | 0.42 |
| D17 | 1.23 | 532 | 42 | 0 | 270 | 34 | 3 | 0.76 | 1.19 | 884 | 100 | 0 | 1777 | 220 | 7 | 5.35 |

Table 9: MC-MDRPP results for instances of the group D16 and D36

| | 2 Depots | | | | | | | | 4 Depots | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
| D18 | 0.64 | 609 | 42 | 0 | 25 | 35 | 1 | 1.04 | 0.05 | 1148 | 88 | 0 | 17 | 16 | 1 | 2.09 |
| D19 | 0.86 | 1321 | 63 | 0 | 217 | 61 | 3 | 1.84 | 3.36 | 1480 | 105 | 0 | 1870 | 202 | 10 | 8.92 |
| D20 | 0.75 | 603 | 40 | 0 | 36 | 28 | 1 | 0.41 | 5.13 | 1106 | 96 | 0 | 765 | 84 | 10 | 2.82 |
| D21 | 0.80 | 1816 | 81 | 0 | 16 | 22 | 1 | 12.87 | 0 | 3487 | 199 | - | - | - | - | 4.60 |
| D22 | 4.42 | 2074 | 94 | 0 | 3346 | 427 | 92 | 42.29 | 2.20 | 2044 | 103 | 0 | 2463 | 277 | 13 | 22.89 |
| D23 | 1.60 | 1051 | 92 | 0 | 252 | 28 | 1 | 3.76 | 1.87 | 2633 | 172 | 0 | 1988 | 83 | 4 | 13.96 |
| D24 | 0.60 | 3073 | 136 | 0 | 273 | 50 | 9 | 8.72 | 1.79 | 13311 | 555 | 0 | 7654 | 11887 | 117 | 435.99 |
| D25 | 1.66 | 1512 | 65 | 0 | 8037 | 680 | 57 | 58.36 | 2.94 | 3104 | 160 | 0 | 5357 | 453 | 74 | 103.49 |
| D26 | 2.95 | 2658 | 112 | 0 | 1445 | 277 | 17 | 26.90 | 3.06 | 7930 | 289 | 0 | 11474 | 888 | 132 | 383.00 |
| D27 | 3.65 | 1755 | 65 | 0 | 4687 | 372 | 25 | 66.42 | 2.18 | 7117 | 250 | 0 | 5617 | 217 | 30 | 150.59 |
| D28 | 2.17 | 2899 | 127 | 0 | 5489 | 270 | 26 | 67.91 | 1.40 | 6023 | 205 | 0 | 3303 | 229 | 11 | 60.79 |
| D29 | 5.03 | 3606 | 123 | 0 | 5177 | 464 | 88 | 127.36 | 3.70 | 7776 | 271 | 0 | 4389 | 263 | 22 | 172.94 |
| D30 | 2.34 | 5257 | 167 | 0 | 2157 | 230 | 6 | 127.89 | 5.72 | 21075 | 513 | 0 | 110867 | 4291 | 145 | 12951.47 |
| D31 | 1.63 | 2799 | 94 | 0 | 2017 | 156 | 8 | 48.11 | 2.38 | 14797 | 458 | 0 | 70606 | 3225 | 318 | 8933.27 |
| D32 | 3.78 | 6158 | 134 | 0 | 18514 | 1163 | 149 | 793.22 | 2.34 | 31960 | 592 | 0 | 46074 | 1569 | 202 | 6728.62 |
| D33 | 2.63 | 7558 | 321 | 0 | 4567 | 624 | 47 | 602.05 | 3.72 | 20318 | 453 | 0 | 45015 | 2191 | 292 | 5967.82 |
| D34 | 4.57 | 6849 | 125 | 0 | 59176 | 3848 | 1284 | 5835.94 | 100 | 4133 | 159 | 100 | 113314 | 3543 | 112 | 14400.02 |
| D35 | 11.17 | 2654 | 64 | 6.34 | 140105 | 6228 | 1757 | 14402.84 | 100 | 47542 | 1420 | 100 | 64354 | 1378 | 38 | 14401.53 |

Table 10: MC-MDRPP results for instances of the group D64 and D100

| | 2 Depots | | | | | | | | 4 Depots | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
| G01 | 0 | 3 | 0 | - | - | - | - | 0.02 | 0 | 0 | 6 | - | - | - | - | 0.01 |
| G02 | 0 | 4 | 10 | - | - | - | - | 0.00 | 0 | 0 | 0 | - | - | - | - | 0.02 |
| G03 | 0 | 18 | 16 | - | - | - | - | 0.00 | 0 | 4 | 0 | - | - | - | - | 0.00 |
| G04 | 0 | 2 | 15 | - | 5 | 6 | 2 | 0.03 | 0 | 6 | 16 | - | - | - | - | 0.01 |
| G06 | 7.50 | 104 | 19 | 0 | 58 | 34 | 6 | 0.09 | 0 | 0 | 0 | - | - | - | - | 0.00 |
| G07 | 0 | 14 | 14 | - | - | - | - | 0.02 | 0 | 4 | 17 | - | - | - | - | 0.00 |
| G08 | 3.13 | 17 | 12 | 0 | 26 | 10 | 2 | 0.03 | 0 | 3 | 12 | - | - | - | - | 0.00 |
| G09 | 0 | 95 | 36 | - | - | - | 2 | 0.05 | 0 | 49 | 32 | - | - | - | 2 | 0.06 |
| G10 | 3.33 | 155 | 38 | 0 | 2 | 5 | 2 | 0.12 | 0 | 195 | 43 | - | - | - | - | 0.09 |
| G11 | 1.56 | 248 | 43 | 0 | 99 | 10 | 2 | 0.23 | 0 | 71 | 46 | - | - | - | - | 0.06 |
| G12 | 2.03 | 547 | 44 | 0 | 227 | 28 | 3 | 0.69 | 0 | 170 | 46 | - | - | - | - | 0.14 |
| G13 | 0.93 | 277 | 45 | 0 | 60 | 20 | 1 | 0.41 | 4.51 | 348 | 54 | 0 | 230 | 39 | 8 | 0.78 |
| G14 | 0 | 263 | 42 | - | - | - | - | 0.11 | 0.83 | 495 | 78 | 0 | 2 | 4 | 1 | 0.61 |
| G15 | 3.87 | 665 | 81 | 0 | 541 | 123 | 5 | 1.79 | 0 | 937 | 154 | - | - | - | - | 1.09 |
| G16 | 3.79 | 456 | 32 | 0 | 776 | 108 | 6 | 1.31 | 8.70 | 683 | 121 | 0 | 6925 | 841 | 400 | 91.60 |
| G17 | 0 | 117 | 28 | 0 | - | - | - | 0.06 | 0.00 | 119 | 54 | - | - | - | - | 0.10 |

Table 11: MC-MDRPP results for instances of the group G16 and G36

| | 2 Depots | | | | | | | | 4 Depots | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | ♯$Nod$ | cpu | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | ♯$Nod$ | cpu |
| G18 | 0 | 67 | 12 | - | - | - | - | 0.03 | 0 | 728 | 110 | - | - | - | - | 0.77 |
| G19 | 1.04 | 716 | 47 | 0 | 33 | 13 | 1 | 0.67 | 0 | 123 | 35 | - | - | - | - | 0.09 |
| G20 | 3.58 | 1053 | 79 | 0 | 511 | 61 | 3 | 2.06 | 0 | 697 | 103 | - | - | - | - | 0.61 |
| G21 | 0.76 | 779 | 60 | 0 | 1772 | 158 | 4 | 6.52 | 1.52 | 3937 | 251 | 0 | 4436 | 337 | 26 | 49.98 |
| G22 | 1.52 | 1003 | 93 | 0 | 10336 | 1039 | 202 | 118.48 | 1.26 | 4782 | 299 | 0 | 44103 | 2825 | 456 | 1816.27 |
| G23 | 0 | 453 | 94 | - | - | - | - | 0.25 | 0 | 538 | 109 | - | - | - | - | 0.98 |
| G24 | 1.67 | 5152 | 208 | 0 | 23600 | 1817 | 210 | 416.18 | 1.67 | 9078 | 520 | 0 | 75960 | 3825 | 598 | 13783.16 |
| G25 | 3.57 | 2769 | 159 | 0 | 27857 | 2432 | 980 | 823.15 | 1.63 | 3611 | 271 | 0 | 21740 | 1394 | 72 | 865.65 |
| G26 | 3.57 | 1282 | 93 | 0 | 5489 | 439 | 77 | 43.62 | 0 | 236 | 74 | - | - | - | - | 0.28 |
| G27 | 3.31 | 2009 | 93 | 0 | 2809 | 193 | 8 | 23.73 | 0.99 | 3113 | 145 | 0 | 664 | 51 | 1 | 11.33 |
| G28 | 1.99 | 4411 | 215 | 0 | 9033 | 361 | 13 | 86.39 | 0 | 8279 | 402 | 0 | - | - | - | 63.59 |
| G29 | 2.24 | 2715 | 131 | 0 | 6850 | 473 | 36 | 81.68 | 2.76 | 10316 | 383 | 0 | 4258 | 239 | 72 | 284.59 |
| G30 | 3.83 | 5569 | 276 | 0 | 35430 | 2421 | 1065 | 2563.3 | 100 | 19182 | 680 | 100 | 86488 | 2316 | 62 | 14400.31 |
| G31 | 2.26 | 1879 | 97 | 0 | 54651 | 2565 | 266 | 1946.39 | 100 | 8586 | 251 | 100 | 97123 | 2703 | 52 | 14400.42 |
| G32 | 0.85 | 30803 | 728 | 0 | 47093 | 2365 | 145 | 4970.55 | 100 | 13823 | 681 | 100 | 87876 | 1810 | 92 | 14400.20 |
| G33 | 9.88 | 52647 | 1264 | 9.88 | 200139 | 4650 | 425 | 14401.01 | 100 | 26911 | 1082 | 100 | 76896 | 554 | 21 | 14400.02 |
| G34 | 15.54 | 17147 | 476 | 15.36 | 153312 | 7559 | 449 | 14400.13 | 100 | 12452 | 649 | 100 | 48598 | 1364 | 11 | 14400.14 |
| G35 | 1.41 | 12140 | 522 | 0 | 29336 | 2262 | 630 | 3206.24 | 100 0 | 12658 | 661 | 100 | 104903 | 3038 | 34 | 14400.36 |

Table 12: MC-MDRPP results for instances of the group G64 and G100

| | 2 Depots | | | | | | | | 4 Depots | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
| R01 | 0 | 0 | 0 | - | - | - | - | 0.02 | 0 | 0 | 4 | - | - | - | - | 0.00 |
| R03 | 0 | 15 | 11 | - | - | - | - | 0.02 | 0 | 3 | 11 | - | - | - | - | 0.03 |
| R05 | 0 | 14 | 0 | - | - | - | - | 0.00 | 0 | 0 | 0 | - | - | - | - | 0.00 |
| R06 | 1.16 | 129 | 24 | 0 | 28 | 34 | 15 | 0.41 | 8.14 | 15 | 21 | 0 | 147 | 37 | 6 | 0.20 |
| R07 | 0 | 22 | 13 | - | - | - | - | 0.03 | 0 | 0 | 6 | - | - | - | - | 0.01 |
| R08 | 0 | 56 | 7 | - | - | - | - | 0.01 | 0 | 42 | 34 | - | - | - | - | 0.16 |
| R09 | 0 | 78 | 14 | - | - | - | - | 0.05 | 0 | 52 | 11 | - | - | - | - | 0.05 |
| R10 | 0 | 229 | 28 | - | - | - | - | 0.06 | 0 | 66 | 13 | - | - | - | - | 0.08 |
| R11 | 0 | 176 | 13 | - | - | - | - | 0.03 | 0 | 90 | 21 | - | - | - | - | 0.06 |
| R12 | 0 | 54 | 0 | - | - | - | - | 0.00 | 0 | 9 | 4 | - | - | - | - | 0.00 |
| R13 | 0.47 | 460 | 33 | 0 | 502 | 90 | 17 | 0.95 | 0.63 | 1110 | 109 | 0 | 100 | 20 | 9 | 1.19 |
| R14 | 0 | 734 | 44 | - | - | - | - | 0.37 | 0 | 488 | 64 | - | - | - | - | 0.42 |
| R15 | 0 | 299 | 23 | - | - | - | - | 0.09 | 0 | 258 | 59 | - | - | - | - | 0.28 |
| R16 | 0 | 889 | 79 | - | - | - | - | 0.25 | 0 | 1212 | 71 | - | - | - | - | 0.41 |
| R17 | 0 | 78 | 15 | - | - | - | - | 0.05 | 0 | 44 | 19 | - | - | - | - | 0.05 |
| R18 | 0 | 250 | 28 | - | - | - | - | 0.08 | 1.17 | 504 | 84 | 0 | 46 | 17 | 2 | 0.61 |
| R19 | 1.75 | 241 | 16 | 0 | 338 | 12 | 2 | 0.37 | 1.26 | 650 | 80 | 0 | 360 | 51 | 2 | 0.91 |

Table 13: MC-MDRPP results for instances of the group R20, R30, R40 and R50

| | 2 Depots | | | | | | | | 4 Depots | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
| P1 | 0 | 25 | 10 | - | - | - | - | 0.02 | 0 | 0 | 0 | - | - | - | - | 0.00 |
| P2 | 6.11 | 66 | 19 | 0 | 115 | 23 | 4 | 0.11 | 0 | 49 | 29 | - | - | - | - | 0.11 |
| P3 | 7.55 | 225 | 19 | 0 | 550 | 140 | 17 | 1.19 | 18.75 | 530 | 82 | 0 | 1102 | 253 | 58 | 6.77 |
| P5 | 16.44 | 164 | 19 | 0 | 231 | 77 | 21 | 0.38 | 44.63 | 330 | 60 | 0 | 198 | 104 | 16 | 1.08 |
| P6 | 3.77 | 361 | 47 | 0 | 964 | 171 | 33 | 1.59 | 23.08 | 736 | 97 | 0 | 1191 | 402 | 133 | 10.56 |
| P10 | 5.13 | 73 | 25 | 0 | 12 | 17 | 8 | 0.16 | 4.17 | 33 | 24 | 0 | 0 | 4 | 1 | 0.09 |
| P14 | 10.55 | 495 | 30 | 0 | 997 | 183 | 56 | 3.48 | 9.09 | 687 | 99 | 0 | 5597 | 980 | 230 | 88.41 |
| P15 | 21.06 | 195 | 16 | 0 | 40 | 41 | 24 | 0.48 | 4.56 | 145 | 36 | 0 | 45 | 25 | 4 | 0.31 |
| P16 | 3.92 | 283 | 22 | 0 | 593 | 25 | 3 | 0.73 | 16.67 | 943 | 86 | 0 | 7267 | 1386 | 734 | 206.86 |
| P17 | 11.67 | 234 | 23 | 0 | 324 | 122 | 75 | 0.97 | 13.83 | 167 | 30 | 0 | 137 | 74 | 5 | 0.50 |
| P19 | 7.87 | 522 | 52 | 0 | 476 | 176 | 49 | 3.59 | 20.20 | 1012 | 101 | 0 | 2139 | 340 | 52 | 14.54 |
| P24 | 5.62 | 1196 | 51 | 0 | 3841 | 902 | 137 | 40.51 | 15.85 | 3310 | 191 | 6.09 | 37267 | 4372 | 12051 | 14400.23 |

Table 14: MM-MDRPP results for instances of the group P

| | 2 Depots | | | | | | | | 4 Depots | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | ♯$Nod$ | cpu | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | ♯$Nod$ | cpu |
| D02 | 0 | 58 | 13 | - | - | - | 0 | 0.06 | 0 | 35 | 30 | - | - | - | 0 | 0.08 |
| D04 | 0 | 36 | 11 | - | - | - | 0 | 0.01 | 0 | 29 | 16 | - | - | - | 0 | 0.03 |
| D05 | 9.11 | 41 | 17 | 0 | 141 | 80 | 21 | 0.20 | 16.03 | 79 | 33 | 0 | 0 | 8 | 5 | 0.19 |
| D06 | 0 | 46 | 22 | - | 25 | - | 0 | 0.06 | 17.56 | 51 | 30 | 0 | 0 | 0 | 2 | 0.12 |
| D07 | 8.26 | 40 | 11 | 0 | - | 23 | 5 | 0.14 | 0 | 39 | 20 | - | - | - | 0 | 0.05 |
| D08 | 6.46 | 196 | 29 | 0 | 45 | 27 | 50 | 0.27 | 0 | 0 | 0 | - | - | - | 0 | 0.00 |
| D09 | 15.23 | 201 | 25 | 0 | 223 | 32 | 13 | 0.44 | 39.47 | 271 | 32 | 0 | 207 | 67 | 26 | 1.05 |
| D10 | 20.11 | 282 | 33 | 0 | 102 | 44 | 6 | 0.27 | 0 | 202 | 54 | - | - | - | - | 0.23 |
| D11 | 19.23 | 272 | 19 | 0 | 2202 | 268 | 60 | 4.90 | 30.25 | 1328 | 103 | 0 | 3648 | 579 | 198 | 30.23 |
| D12 | 15.94 | 390 | 34 | 0 | 414 | 96 | 24 | 1.11 | 35.32 | 602 | 58 | 0 | 1640 | 331 | 143 | 7.78 |
| D13 | 9.29 | 388 | 37 | 0 | 1149 | 205 | 45 | 2.14 | 14.39 | 650 | 105 | 0 | 288 | 90 | 21 | 3.15 |
| D14 | 11.85 | 577 | 76 | 0 | 3379 | 414 | 135 | 18.82 | 23.90 | 1669 | 139 | 0 | 7215 | 1333 | 445 | 157.48 |
| D15 | 9.79 | 674 | 47 | 0 | 578 | 117 | 23 | 2.03 | 21.78 | 1625 | 183 | 0 | 6792 | 894 | 433 | 84.93 |
| D16 | 8.15 | 511 | 52 | 0 | 1220 | 179 | 38 | 4.35 | 5.34 | 1546 | 195 | 0 | 625 | 134 | 13 | 8.669 |
| D17 | 3.70 | 744 | 53 | 0 | 2527 | 383 | 56 | 9.89 | 10.06 | 2492 | 232 | 0 | 8111 | 989 | 310 | 129.73 |
| D18 | 10.23 | 1404 | 47 | 0 | 5315 | 337 | 35 | 12.53 | | | | | | | | |
| D19 | 9.60 | 2025 | 77 | 0 | 5985 | 540 | 64 | 20.53 | | | | | | | | |
| D20 | 28.67 | 659 | 29 | 0 | 6048 | 912 | 678 | 48.00 | | | | | | | | |
| D21 | 7.44 | 1980 | 93 | 0 | 15120 | 964 | 198 | 136.13 | | | | | | | | |
| D22 | 8.32 | 1947 | 74 | 0 | 19848 | 1487 | 273 | 258.9 | | | | | | | | |
| D23 | 11.92 | 1761 | 172 | 0 | 9470 | 1126 | 187 | 444.13 | | | | | | | | |
| D24 | 6.64 | 3235 | 92 | 0 | 24516 | 1848 | 697 | 580.87 | | | | | | | | |
| D25 | 5.79 | 2953 | 92 | 0 | 17770 | 1165 | 221 | 894.59 | | | | | | | | |
| D26 | 12.33 | 2799 | 119 | 1.02 | 51991 | 4979 | 12843 | 14400 | | | | | | | | |

Table 15: MM-MDRPP results for instances of the group D16, D36, and D64

| | 2 Depots | | | | | | | | 4 Depots | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | $\sharp Nod$ | cpu |
| G01 | 24.00 | 27 | 11 | 0 | 6 | 14 | 8 | 0.05 | 22.22 | 0 | 10 | 0 | 0 | 0 | 1 | 0.01 |
| G02 | 0 | 3 | 7 | - | - | - | - | 0.00 | 0 | 0 | 4 | - | - | - | - | 0.00 |
| G03 | 0 | 8 | 8 | - | - | - | - | 0.02 | 0 | 0 | 0 | - | - | - | - | 0.00 |
| G04 | 24.00 | 37 | 12 | 0 | 72 | 37 | 10 | 0.06 | 0 | 30 | 37 | - | - | - | - | 0.05 |
| G06 | 0 | 85 | 19 | - | 16 | 12 | 1 | 0.06 | 0 | 0 | 0 | - | - | - | - | 0.00 |
| G07 | 0 | 58 | 30 | - | 1 | 2 | 1 | 0.05 | 0 | 50 | 34 | - | - | - | - | 0.05 |
| G08 | 20.00 | 51 | 10 | 0 | 34 | 26 | 6 | 0.06 | 0 | 3 | 9 | - | - | - | - | 0.01 |
| G09 | 0 | 157 | 22 | - | 10 | 12 | 1 | 0.13 | 0 | 74 | 41 | - | - | - | - | 0.13 |
| G10 | 6.25 | 186 | 24 | 0 | 312 | 117 | 8 | 0.58 | 0 | 345 | 53 | - | - | - | - | 0.20 |
| G11 | 11.11 | 236 | 43 | 0 | 106 | 15 | 2 | 0.28 | 32.19 | 401 | 75 | 0 | 507 | 148 | 13 | 1.83 |
| G12 | 9.09 | 655 | 54 | 0 | 1949 | 307 | 33 | 6.44 | 16.67 | 1631 | 189 | 0 | 6168 | 1345 | 457 | 149.26 |
| G13 | 5.00 | 279 | 42 | 0 | 197 | 62 | 2 | 0.47 | 28.57 | 614 | 98 | 0 | 835 | 311 | 94 | 7.24 |
| G14 | 4.55 | 386 | 71 | 0 | 952 | 180 | 17 | 2.46 | 28.05 | 565 | 94 | 0 | 4161 | 1053 | 883 | 130.60 |
| G15 | 7.69 | 974 | 67 | 0 | 3976 | 624 | 71 | 18.71 | 15.17 | 1461 | 277 | 0 | 7023 | 1077 | 429 | 221.77 |
| G16 | 11.36 | 359 | 26 | 0 | 4014 | 704 | 116 | 18.11 | 8.33 | 1242 | 104 | 0 | 11115 | 1498 | 407 | 170.79 |
| G17 | 0 | 185 | 45 | - | - | - | - | 0.11 | 14.10 | 919 | 192 | 0 | 3151 | 686 | 374 | 9372.00 |
| G18 | 21.99 | 435 | 43 | 0 | 4181 | 562 | 146 | 20.73 | 22.95 | 28 | | | | | | |
| G19 | 9.11 | 653 | 46 | 0 | 1920 | 173 | 12 | 3.13 | 23.83 | 26 | | | | | | |
| G20 | 11.91 | 785 | 52 | 0 | 7761 | 1128 | 191 | 63.74 | 25.02 | 28 | | | | | | |
| G21 | 10.20 | 1212 | 59 | 0 | 27044 | 2515 | 1420 | 1173.03 | 32.67 | 36 | | | | | | |
| G22 | 3.03 | 2398 | 78 | 0 | 10047 | 684 | 33 | 86.14 | 33.00 | 34 | | | | | | |
| G23 | 5.26 | 3318 | 180 | 0 | 30841 | 2667 | 468 | 1578.73 | 38.00 | 40 | | | | | | |
| G24 | 3.95 | 10772 | 375 | 0 | 47915 | 3719 | 412 | 2764.37 | 44.25 | 46 | | | | | | |
| G25 | 2.44 | 3466 | 172 | 0 | 49735 | 6175 | 2046 | 6426.29 | 41.00 | 42 | | | | | | |
| G26 | 2.44 | 1444 | 97 | 0 | 51206 | 5165 | 893 | 2869.26 | 41.00 | 42 | | | | | | |

Table 16: MM-MDRPP results for instances of the group G16 and G36

| | 2 Depots | | | | | | | | 4 Depots | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | ♯$Nod$ | cpu | $gap_0$ | $cutsC_0$ | $cutsP_0$ | gap | cutsC | cutsP | ♯$Nod$ | cpu |
| R01 | 0 | 3 | 2 | - | - | - c | - | 0.00 | 0 | 0 | 4 | - | - | - | - | 0.00 |
| R03 | 26.23 | 33 | 8 | 0 | 111 | 47 | 14 | 0.31 | 0 | 39 | 28 | - | - | - | - | 0.09 |
| R05 | 0 | 19 | 2 | - | - | - | - | 0.05 | 0 | 0 | 0 | - | - | - | - | 0.00 |
| R06 | 22.04 | 174 | 21 | 0 | 5 | 8 | 1 | 0.12 | 9.59 | 253 | 47 | 0 | 6 | 6 | 4 | 0.23 |
| R07 | 3.70 | 45 | 19 | 0 | 28 | 6 | 5 | 0.14 | 0 | 36 | 19 | - | - | - | - | 0.09 |
| R08 | 22.41 | 195 | 25 | 0 | 429 | 40 | 9 | 0.38 | 0 | 276 | 66 | - | - | - | - | 0.51 |
| R09 | 1.87 | 128 | 15 | 0 | 0 | 0 | 2 | 0.06 | 24.04 | 156 | 19 | 0 | 404 | 115 | 27 | 1.08 |
| R10 | 1.05 | 239 | 19 | 0 | 24 | 7 | 2 | 0.20 | 32.44 | 624 | 50 | 0 | 1746 | 244 | 81 | 5.87 |
| R11 | 0 | 231 | 17 | - | - | - | - | 0.08 | 49.30 | 139 | 23 | 0 | 24 | 37 | 6 | 0.48 |
| R12 | 0 | 133 | 11 | - | - | - | - | 0.08 | 46.16 | 96 | 28 | 0 | 80 | 66 | 19 | 0.80 |
| R13 | 9.76 | 696 | 36 | 0 | 1639 | 217 | 173 | 5.84 | 14.50 | 1209 | 109 | 0 | 8113 | 952 | 405 | 131.82 |
| R14 | 6.55 | 792 | 34 | 0 | 1632 | 322 | 120 | 5.88 | 13.05 | 1123 | 99 | 0 | 1889 | 233 | 89 | 12.29 |
| R15 | 12.83 | 630 | 22 | 0 | 7620 | 542 | 165 | 14.81 | 14.55 | 647 | 65 | 0 | 4967 | 812 | 619 | 51.07 |
| R16 | 9.83 | 593 | 46 | 0 | 756 | 59 | 6 | 0.67 | 57.87 | 1596 | 125 | 0 | 9195 | 2921 | 2589 | 361.14 |
| R17 | 44.08 | 238 | 15 | 0 | 545 | 85 | 30 | 0.48 | 35.56 | 162 | 32 | 0 | 176 | 46 | 3 | 0.41 |
| R18 | 15.91 | 392 | 40 | 0 | 1184 | 131 | 28 | 1.15 | 46.89 | 1108 | 126 | 0 | 8216 | 1340 | 705 | 105.69 |
| R19 | 9.81 | 837 | 3 | 0 | 3643 | 171 | 50 | 3.34 | 29.04 | 1515 | 116 | 0 | 6489 | 644 | 160 | 50.72 |

Table 17: MM-MDRPP results for instances of the group R20, R30, R40, and R50